

Appendix A – Listing of Claims After Cancellations in Amendment After Allowance under 35 USC 1.312

1. A method for monitoring thread usage in a server system, comprising:
sending an ioctl call in blocking mode on a socket designated for listening for incoming client requests to a server communicatively connected to a network and passing said incoming client requests to one from among a plurality of threads waiting in a thread pool;
responsive to a TCP layer detecting said listen socket in blocking mode, monitoring a thread count of at least one of a number of incoming requests waiting to be processed and a number of said plurality of threads remaining idle in said thread pool over a sample period; and
responsive to said TCP layer detecting a thread usage event, returning said ioctl call back with said thread count, such that a number of threads in said thread pool is dynamically adjusted to handle said thread count.
2. The method according to claim 1 for monitoring thread usage wherein said monitoring a thread count further comprises setting a counter to monitor said number of incoming requests waiting to be processed over a particular number of TCP slow timer processing cycles.
3. The method according to claim 1 for monitoring thread usage wherein said monitoring a thread count further comprises monitoring a minimum number of said number of threads remaining idle over said sample period.
4. The method according to claim 1 for monitoring thread usage further comprising:
dynamically adjusting a number of active threads in said thread pool according to said thread count to handle a current load.

5. The method according to claim 1 for monitoring thread usage further comprising:

initiating a TCP slow timer cycle;

processing all of a plurality of sockets during said TCP slow timer cycle;

responsive to completing said processing of all of said plurality of

sockets, comparing said thread count with a threshold;

responsive to said thread count exceeding said threshold, designating a thread usage event with said number of incoming requests waiting to be processed; and

responsive to said thread count equaling zero after said sample period, designating a thread usage event with said number of threads remaining idle.

16. A method for monitoring thread usage to dynamically adjust a number of active threads in a thread pool of a server system, comprising:

sending an ioctl call in blocking mode on a socket designated for listening for incoming client requests to a server system communicatively connected to a network and passing said incoming client requests to one from among a plurality of active threads waiting in a thread pool;

responsive to a TCP layer of said server system detecting a thread usage event, receiving said ioctl call back with a thread count of at least one of a number of incoming requests waiting to be processed and a number of said plurality of threads remaining idle in said thread pool over a sample period; and

dynamically adjusting said number of active threads in said thread pool according to said thread count, such that said server system dynamically adjusts said thread pool to handle a current load.